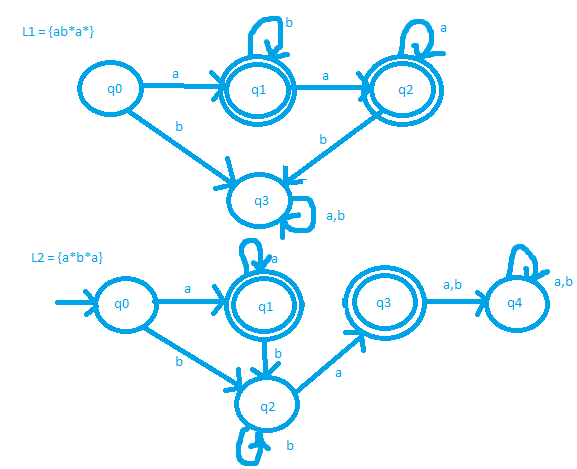
CSci 435: Formal Languages and Automata

Instructor: Dr. M. E. Kim Name: \_Sam Dressler\_\_\_\_\_\_\_\_\_\_

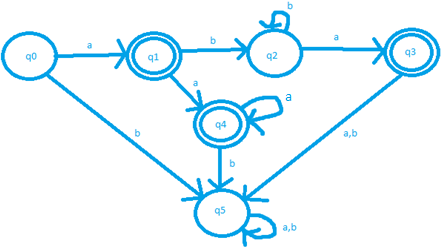
**Home Assignment 3: 92/100 points + 25 points (optional)**

Q1. [15/10]

1. Use the construction in Theorem 4.1 to find an DFA that accept L(*ab\*a*\*) ∩ L(*a\*b\*a*).



DFA for L1 ∩ L2

 very good

1. [5, Optional] Give the regular expression for the above language in 1) that is accepted by your DFA.

L = {(a(a\*)) + (ab\*a)}

\*Q2. [10] The ***complementary or (cor)*** of two sets L1 and L2 is defined as

cor(L1, L2) = {*w* | *w* ∉L1 or *w* ∉ L2, }.

Show that the family of regular languages is ***closed*** under ***cor.***

To show that the family of regular languages is closed under cor, assume L­1 and L2 are Regular Languages.

Then there will be a DFA M1 = (Q, ∑, δ, q0, F) that accepts L1. So, L(M1) = L1

and a DFA M2 = (Q, ∑, δ, q0, F) that accepts L­2. So, L(M­2) = L2

There must also exist DFA’s M1’ and M2’ such that L(M1’) = L1’ and L(M2’) = L2’

We can construct a DFA M1’ with (Q, ∑, δ, q0, Q-F) that accepts the complement of L1, L1’.

We can also construct a DFA M­2’ = (Q, ∑, δ, q0, Q-F) that accepts the complement of L­2, L2’.

Since L1’ and L2’ accept strings that L1 and L2 reject and vice versa, L1’ and L2’ are regular languages.

Cor(L1, L2) can be rewritten to accept {w | w∈ L1’ U L2’} by theorem 4.1

Therefore, the family of languages (L1, L­2), which accepts strings from L1’ or L2’, is closed.

Since *w* ∉L1 or *w* ∉L1 , w ∈ or w ∈ .

Thus, cor(L1, L2) = { w | w ∈ or w ∈ } = ∪ .

The result then follows from closure under union and complement of regular language.

\*Q3. [0/ 10] The family of regular languages are closed under arbitrary ***homomorphism***.

Prove or disprove h(L1 ∩ L2) =h(L1) ∩ h(L2) is a regular language where L1 and L2 are regular.

Prove that h (L1 ∩ L­2) = h (L1) ∩ h (L2)

By theorem 4.3, if L is a regular language, then h(L) is regular.

So, h(L­1) and h(L2) are regular.

By theorem 4.1, if L­1 and L2 are regular languages then L1 ∩ L2 is regular.

Let L3 = L1 ∩ L2

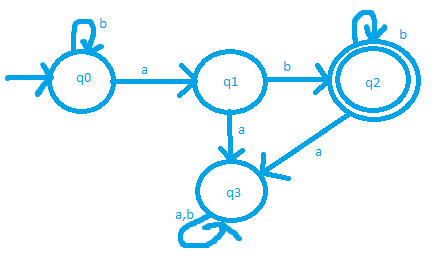
L3 is regular by 4.1, then by 4.3 h(L3) is regular, thus h(L1 ∩ L2) is a regular language.

False. For example,

L1 = L(*a*\*), L2 = L(b\*), h(*a*) = *a* and h(b) = *a*, then, h(L1 ∩ L2) = ∅.

But, h(L1) ∩ h(L2) = L(*a*\*).

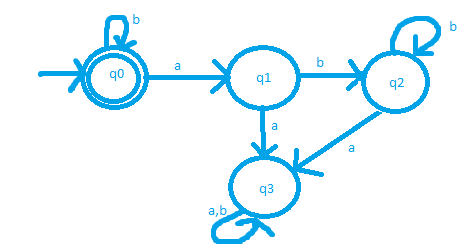
Q4. [10] Let L1 = {L(*b*\**abb*\*) and L2 = L(*bab*\*). Find the ***right quotient*** of L1 with L2, L1/L2.

DFA for L1

1. [5] Let M be a DFA s.t. L(M) = L(L1). By applying Thm. 4.4, construct a DFA M’ s.t. L(M’) = L1/L2.

To find the right quotient of L­1/L2 , check each state to see if it accepts the string defined by the regular expression in L2.

Only q­0 works so the final state in M’ will be q0



1. [5] Then, give a regular expression for L(M’) = L1/L2.

The Regular expression of the regular language given by L1/L2 is obvious when looking at the DFA.

Therefore, the regular expression for L(M’) = L(L1L2 ) = {b\*}

\*Q5. [10] If L is a regular language, prove that the language L2 = { *uv* | *u*∈ LR , *v* ∈L } is also regular.

Show that the language L2 is regular.

Since L is regular, by theorem 4.2, so is the reversal LR.

Then the strings accepted by L2 accepts the subset LR⋅L, thus, L2 is regular language by theorem 4.1

\*Q6. [10] The ***left quotient*** of a regular language L1 with respect to L2 is defined as:

L2/L1 = { *y* | *x*∈ L2 , *xy* ∈L1 }

*Show that the family of regular languages is* ***closed*** *under the* ***left quotient*** *with a regular language*.

Hint: Do NOT construct a DFA that accepts L2/L1 but use the definition of L2/L1 and the closure

properties of regular language.

L2/L1 = “left quotient of L1 with L2”

To show the family of regular languages is closed under the left quotient with a regular language:

We can take the reverse of the left quotient: (L2/L1)R = {yR | yRxR ∈ L1R, xR ∈ L2} = L1R/L­2R

Since L1 and L2 are regular, L1R andL­2R are regular by theorem 4.2

So, the right quotient L1 with L2: L1R/ L2R is regular.

This can be rewritten as (L2/ L1)R

Then ((L2/ L1) R) R is regular by Theorem 4.2, therefore L2/L1 is regular.

? Q7. [7/ 10] Disprove that L1 = L1L2/L2 for all languages L1 and L2 . Give a counter example.

Let L­1­L2 = {xy | x ∈ L1, y ∈ L2} and L1/L2 = {x | xy ∈ L1 for some y ∈ L2} where L1 and L2 are regular languages.

Example: **Let L1 = {anbn|n >=0}** & **L2 = {bn | n >= 0}**

So L1L2 = {xy | x ∈ L1, y ∈ L2} Or L1L2 = {an bn bn| n>=0}

Then L1L2/L2 = {an | n>=0} not necessarily L1L2/L2 = { *anbm* | *n≠ 0, n > m* ≥ 0 }

Therefore L1 != L1L2/L2 since the language L1 has even numbers of a’s and b’s but L1L2/L2 has no ‘b’s

Therefore, the language L1 is not closed for all L1 and L2

\*Q8. [10] A language is said to be a ***palindrome*** language if L = LR. (4.2-3)

Show that there exists an ***algorithm*** for determining if a given regular language is a palindrome language.

1. Create a DFA, D for L
2. Create an NFA, N that accepts LR
3. Convert N into an equivalent DFA D0 , L(D0) = LR
4. By following the equivalence algorithm defined by theorem 4.7, you can determine if L = LR, or if L is a palindrome.

\*Q9. [20] Pumping Lemma

1. [10] Prove that the language L = {*anbkcn* | *n* ≥ 0, *k* ≥ *n* } is ***not regular***.

A language is regular if the following conditions are satisfied:

1. |xy| <= m

2. |y| >= 1

3. xyiz is included in the language for i >= 0

Start by assuming that L is regular

Next form a string anbkcn with some n>= 0 and k >= n

Let n = 2 and k = 3, so s = a2b3c2

s = aabbbcc and |s| = 7 = m

Divide s into three parts: x, y, and z.

**Let x = a, y = abbb, and z=cc**

To satisfy 1, |xy| must be less than m. Let p = |xy| = 5,

since p <= m or 5 <= 7, condition 1 is satisfied.

To satisfy 2, |y| must be >= 1.

Since |y| = 4, this condition is met.

Finally, to satisfy 3, xyiz is included in the language for i >= 0

For i = 2, xy2z = aabbbabbbcc

Therefore, s is not included in L and condition 3 is not met.

L cannot be regular since condition 3 is not met.

1. [10, Optional] Prove that the language L = {*w* | *na*(*w*) ≠ *nb*(*w*)} is ***not regular***.

L = {a, aab, abb, abbb, aabbb, …}

A language is regular if the following conditions are satisfied:

1. |xy| <= m

2. |y| >= 1

3. xyiz is included in the language for i >= 0

Assume L is regular, therefore there is a string s = aabbb (na(s) =2) ≠ (nb(w) = 3)

s= aabbb, so |s| = 5 = m

divide s into three parts: x, y, z.

**Let x = a, y = abb, z = b**

To satisfy 1, |xy| must be less than m. Let p = |xy| = 4,

since p <= m, condition 1 is met.

To satisfy 2, |y| must be >= 1.

Since |y| = 3, this condition is met.

Finally, to satisfy3, xyiz must be included in the language for i >= 0.

For i = 0:

xyiz = ab and thus does not fit the condition na(w) ≠ nb(w)

Language L can not be regular since condition 3 is not met.

1. [0/ 10] Prove or disprove that L1 ∪ L2 is not regular language if L1 and L2 are not regular languages.

If L1 and L2 are is not both regular languages we can see from theorem 4.1 that L1 U L2 is not regular.

False.

Counter Example: Let L1 = { *anbm* | *n* ≤ *m* } and L1 = { *anbm* | *n* > *m* }.

Both L1 and L2 are not regular.

But, L1 ∪ L2 = L(*a*\**b*\*) is regular.

Q10 [10, optional] The ***min*** of a language L is defined as

***min***(L) = { *w* ∈L | there is no *u* ∈L, *v*∈Σ+, such that *w* = *uv* }.

Show that the family of regular languages is closed under the ***min*** operation.